Empirical Evaluation of Virtual Machine Migration Policies on Power- and Time-Management in Cloud Computing

Kalim Qureshi*, Abdulatif Albusairi, and Paul Manuel

Department of Information Science, College of Life Sciences, Kuwait University, Kuwait
Email: kalimuddin.qureshi@ku.edu.kw (K.Q.); latif00@yahoo.com (A.A.); p.manuel@ku.edu.kw (P.M.)
*Corresponding author

Manuscript received June 30, 2024; revised August 13, 2024; accepted January 8, 2025; published May 8, 2025

***Abstract*—Cloud computing has become a norm for enterprises due to its significant advantages in infrastructure management, performance, and expenses. Cloud data centers consume significant power and go against the principles of green computing. It requires efficient management to minimize the environmental impact. Thus, green computing has become an interesting field of research in cloud computing. However, green computing is bundled with a performance-energy trade-off: job completion rate versus power consumption rate. This paper precisely focuses on this topic. A graphical user interface that utilizes the CloudSim simulator to evaluate the performance of various Virtual Machine (VM) power management policies in data centres has been developed. In our study, we test seven VM allocation policies: Dynamic Voltage and Frequency Scaling (DVFS), Interquartile Range (IQR), Local Regression (LR), Local Regression Robust (LRR), Median Absolute Deviation (MAD), Static Threshold (THR), and Single Threshold (STR). We also evaluate the performance of five VM selection policies: None, Maximum Correlation (MC), Minimum Migration Time (MMT), Maximum Utilization (MU), and Random Selection (RS). Utilizing CloudSim's thirty-six different power management mechanisms across seven designed scenarios, we measure each policy's power consumption and job completion rates for the analysis. The DVFS mechanism has proven to be the most effective method for conserving power. While preserving the power, its job completion rate is significantly compromised. Our proposed model measures the excessive power consumed by a power management mechanism residing in the system and concludes that there is no clear leader based on the performance-energy trade off.**

***Keywords*—green computing, data center, cloud computing, Virtual Machine (VM) migration, green computing trade-off**

I. Introduction

Power-efficient resource management is crucial for distributed cloud data centers' economic and environmental sustainability, which consume enormous amounts of electric energy while delivering various services. Virtual Machine (VM) consolidation is a powerful tool that ensures the provision of services without compromising QoS with a lesser active physical server. To address this issue, this study aims to evaluate different power management mechanisms using computer simulations with the CloudSim toolkit.

Therefore, keeping the server underutilized and inactive is a crime from a power consumption perspective. In large data centers, reducing energy consumption benefits cloud service providers and users and decreases CO_2 emissions [1, 2]. Virtual machine consolidation is a process that aims to reduce energy consumption by optimizing resource utilization. In a cloud data center, migrating all or a few VMs can balance host utilization. Thus, by migration, VMs can consolidate into fewer hosts. This way, the underutilized host will be turned off, or the overutilized host will be converted into an average host. Thus, proper resource utilization can reduce energy consumption. Effective VM management is critical in cloud data centers to rapidly add new Physical Machines (PMs) and remove old, failed, and corrupted ones [3].

While cloud computing focuses on the efficient delivery of hosted computing services, green computing focuses on minimizing the hazardous impact of cloud infrastructures on the environment. The research question is how to assess the trade-offs between energy consumption and computing efficiency of cloud resources [4–6].

In Fig. 1, we can see the process of VM consolidation. In the beginning, the VMs are spread out across different hosts. Then, they are moved from underutilized hosts to others that can handle them, reducing the number of active hosts. Finally, hosts not running active VMs are turned off to save energy. This process is called Virtual Machine Consolidation (VMC) Techniques to Reduce Energy [4, 5]. VMC can be classified as either static or dynamic consolidation. Static VMC is used when a pre-informed requirement about the VM is available, while dynamic VMC is used for on-demand VM provisioning through migration [6, 7]. There are two ways to achieve Virtual Machine consideration [3]:

1) Server virtualization is a technology that partitions the physical machine into multiple Virtual Machines (VMs), each capable of running applications like a physical machine [8].
2) Virtual Machine (VM) migration is a powerful management technique that gives data center operators can adapt the placement of VMs to satisfy performance objectives, improve resource utilization and communication locality, mitigate performance hot spots, achieve fault tolerance, reduce energy consumption, and facilitate system maintenance activities [8]. Fig. 1 illustrates the distribution of VMs before and after consolidation. Initially, the VMs are bound with three hosts, but after consolidation, they are combined on Host 3, while Hosts 1 and 2 are turned into sleep mode, resulting in reduced energy consumption.

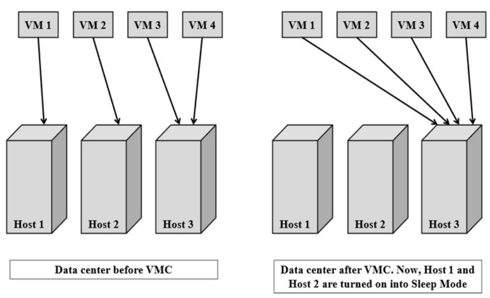

Fig. 1. Virtual machine consolidation.

This work makes a significant contribution by empirically evaluating the performance of VM policies in cloud computing. Improving energy efficiency while maintaining the Quality of Service (QoS) is a highly challenging task.

II. Power Management

Following the power management policies were investigated:

1) Dynamic Voltage and Frequency Scaling (DVFS): The Dynamic Voltage and Frequency Scaling technology adjusts hardware power consumption according to the applied computing load [4].
2) Static Threshold (THR): The Virtual Machine (VM) allocation process is static and does not adapt at run-time, meaning that no other resources except for the CPU are taken into account during VM reallocation [5]. To address this, upper and lower utilization thresholds can be set for hosts to maintain the total CPU utilization of all VMs between these thresholds. If the CPU utilization of a host falls below the lower threshold, all VMs must be moved from that host, and the host should be switched to sleep mode to eliminate idle power consumption. Conversely, if the utilization exceeds the upper threshold, some VMs should be migrated from the host to reduce utilization and prevent potential SLAVs [6, 7].
3) Median Absolute Deviation (MAD): The MAD algorithm adjusts the upper utilization threshold based on the CPU utilization deviation. It uses the Median Absolute Deviation (MAD), a statistical dispersion measure. Once the MAD is calculated, a new threshold is determined to modify the VM migration policy. This approach helps ensure system resources are utilized more efficiently [9].
4) Interquartile Range (IQR): The Interquartile Range (IQR), also called the mid-spread or middle fifty, is a measure of statistical dispersion, being equal to the difference between the third and first quartiles [10].
5) Local Regression (LR): The LR method involves fitting simple models to localized subsets of data to build up a curve approximating the original data [11].
6) Local Regression Robust (LRR): This is a modification of the LR method that eliminates outliers [11].
7) Single Threshold (STR): Measurement threshold policies enable you to monitor performance metrics from various sources. You can configure policies to create events and launch commands whenever a performance metric crosses a threshold [12].

III. Virtual Machine Selection Policies

In this section, describe the VM selection policies that were investigated.

1) The Minimum Migration Time (MMT) policy: The Minimum Migration Time (MMT) policy is to migrate a Virtual Machine (VM) that requires the minimum time to complete a migration compared to other VMs allocated to the host. The estimated migration time is calculated based on the amount of RAM utilized by the VM and divided by the spare network bandwidth available for the host. This policy is designed to ensure efficient resource use during VM migration. The reference for this information is [13].
2) The Random Selection (RS) policy: The RS policy selects VMs according to a uniformly distributed discrete random variable. The virtual machine undergoing migration is chosen using a uniformly distributed discrete random variable [14].
3) The Maximum Correlation (MC) policy: The MC policy selects those VMs with the highest CPU utilization correlation with other VMs. According to this policy, if there is a high correlation between the resource usage of applications running on an oversubscribed server, the likelihood of server overloading increases. Therefore, the Virtual Machines (VMs) that will be migrated have the most significant correlation with the CPU utilization of other VMs. This principle is outlined in reference [15].
4) The minimum Utilization (MU) policy: Here, utilization focuses on power consumption. In this policy, the system identifies the host with the lowest utilization among the hosts and attempts to migrate the VMs from the overburdened host to other underutilized hosts to prevent overloading [15].

The block diagram of the proposed investigation is shown in Fig. 2.

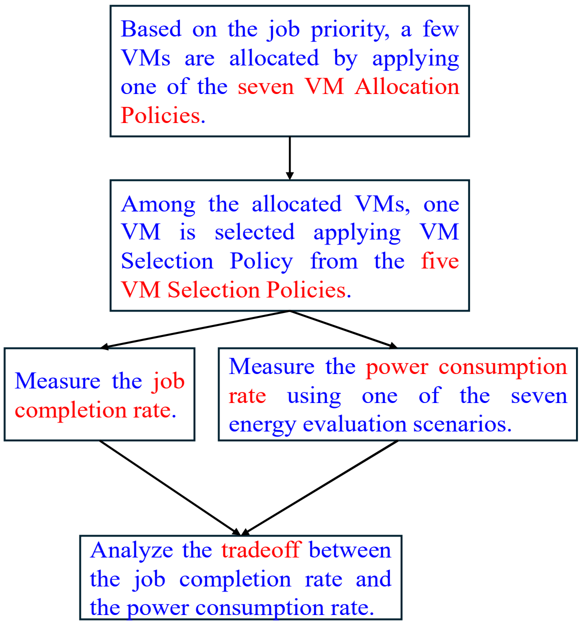

Fig. 2. Block diagram of proposed investigation.

IV. Performance Measurement Methodology

To simplify experiment design in CloudSim, we will create a graphical interface for running and evaluating different load mechanisms. The user interface allows the following functions:

1) Specify the experiment parameters.
2) Apply the parameters and commit them to memory.
3) Save the experiment parameters to disk.
4) Load the experiment parameters to disk.
5) Simulate with the current parameters.
6) View the output.
7) Copy the output to the clipboard.

Table 1 lists seven VM Allocation Policies, and Table 2, the five VM Selection Policies for CloudSim. The platform also allows simulation without power awareness.

Table 1. VM allocation policies

#	List of Investigated Policies
1	Dynamic Voltage and Frequency Scaling (DVFS)
2	Inter Quartile Range (IQR)
3	Local Regression (LR)
4	Local Regression Robust (LRR)
5	Median Absolute Deviation (MAD)
6	Static Threshold (THR)
7	Single Threshold (STR)

Table 2. VM selection policies

#	List of Investigated Selection Policies
1	None
2	Maximum Correlation (MC)
3	Minimum Migration Time (MMT)
4	Maximum Utilization (MU)
5	Random Selection (RS)

V. Implementation of Experiments

To be able to run the experiments, a tool was created to capture the experiments' parameters, execute multiple runs, and record the readings. Many variables or parameters could be changed, but we manipulated only three:

1) The number of hosts.
2) The number of virtual machines.
3) The number of cloudlets (tasks).

A total of seven scenarios were designed. Table 3 below shows the scenarios that were created.

Table 3. Scenario design

Scenario	Hosts	VMs	Cloudlets
1	Increase	Increase	Increase
2	Increase	Increase	-
3	Increase	-	Increase
4	Increase	-	-
5	-	Increase	Increase
6	-	Increase	-
7	-	-	Increase

As shown in Table 4, selecting a power mechanism offers 36 possibilities for running the simulations. For example, choose one VM election policy to evaluate seven VM allocation policies. In this way, thirty-five experiments were added and one non-power-aware; therefore, thirty-six experiments were organized.

Table 4. Power mechanisms tested

Experiment No.	VM Allocation Policy	VM Selection Policy
1–7	DVFS / IQR / LR / LRR / MAD / THR / STR	MC
8–14	DVFS / IQR / LR / LRR / MAD / THR / STR	MMT
15–21	DVFS / IQR / LR / LRR / MAD / THR / STR	MU
22–28	DVFS / IQR / LR / LRR / MAD / THR / STR	RS
29–35	DVFS / IQR / LR / LRR / MAD / THR / STR	None
36	Non-Power-Aware	

Table 5 shows the experimental evaluation configuration. For example, scenario one is evaluated by 10 experiments with different hosts, VMs, and Cloudlets configurations.

Each Scenario was run against each power mechanism for 10 experiments for Scenarios 1–7. Therefore, here, we explicitly define each scenario and describe the measured data.

Table 5. Scenario parameters (Scenarios 1–7 are applied to each experiment)

Experiment	Hosts	VMs	Cloudlets
1	100	100	100
2	200	200	200
3	300	300	300
4	400	400	400
5	500	500	500
6	600	600	600
7	700	700	700
8	800	800	800
9	900	900	900
10	1000	1000	1000

A. Scenario 1: Increasing the No. of Hosts, VMs, and Cloudlets

In this scenario, we varied the number of hosts, VMs, and Cloudlets and observed the power consumption and job completion time. The objective was to observe the effect of each parameter variation.

The measured data of power consumption for experiments 1–10 is shown in Fig. 3:

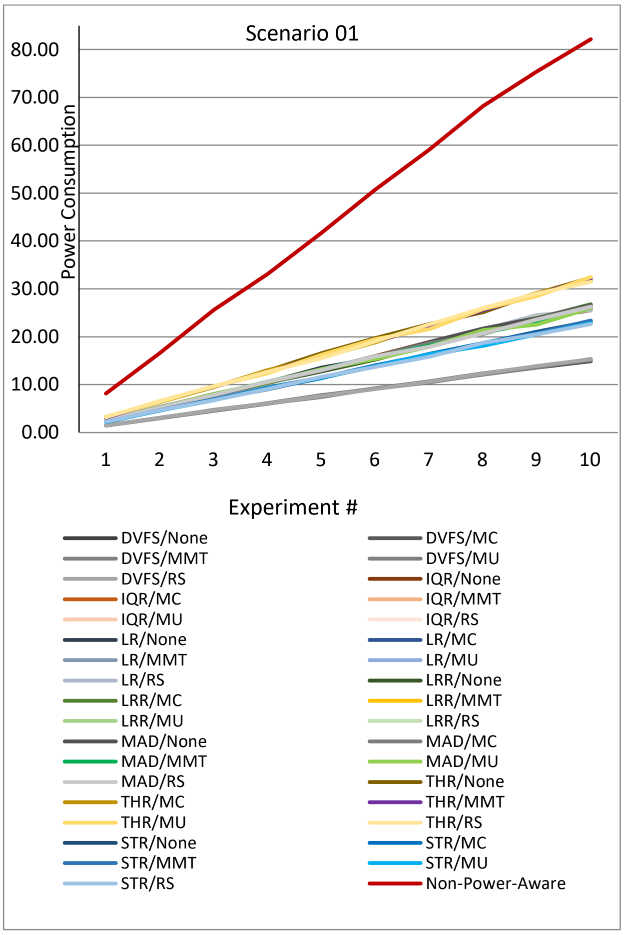

Fig. 3. Scenario 1: Power consumption for Experiment #1.

1. The measured result observes that a strategy without power awareness always requires more power (80 W) than all other investigated policies.
2. DVFS, with the combination of all VM selection policies, takes a minimum power consumption (1–10 W) only.
3. All other combinations take a power consumption of 11–30 W, which is high.

The measured data of job completion time for experiments 1–10 is shown in Fig. 4:

1. The minimum time measured by TFR/MMT (1174.39 s in experiment # 10) is close to THR/MU (1174.39 s) LRR/MU (1175.81 s).
2. Maximum time taken to complete the job is STR/MMT (1225.10 s) and THR/MC (1223.67 s).
3. The time taken by non-power-aware is 1190.10 s, which is higher than the minimum time measured by TFR/MMT.

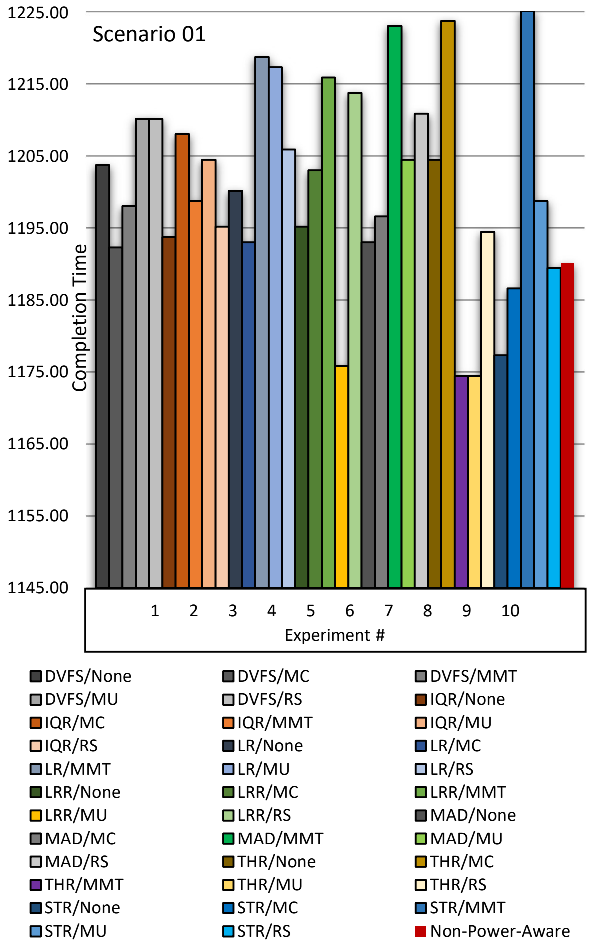

Fig. 4. Scenario 1: Experiment #1 for job completion time.

B. Scenario 2: Increasing the No. of Hosts and Number of VMs

In this scenario, we varied the number of hosts and VMs while keeping the number of Cloudlets constant and observed the power consumption and job completion time. The objective of this scenario is to observe the effect of the variation in the number of hosts and VMs. The results for the power consumption and job completion time are shown in Figs. 5 and 6.

The measured results from Fig. 5 show that in all 10 experiments, the DVFS power mechanisms had the least power consumption (17 W in experiment 10). The non-power-aware took a high-power requirement in all experiments (87 W in experiment 10).

Fig. 6 shows that the STR/None had the minimum execution time in all experiments. Even though the non-power-aware had a competitive job completion time (1188 s), its power consumption was the worst, as it kept all the servers powered on.

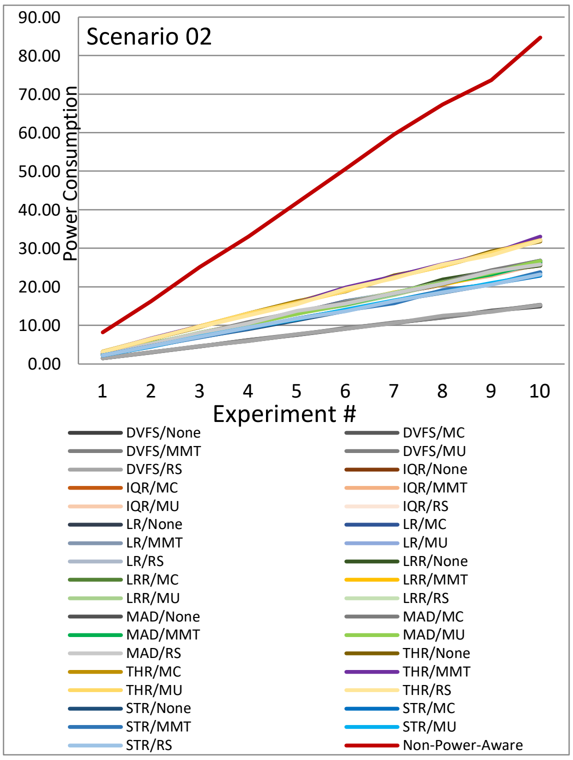

Fig. 5. Scenario 2: Power consumption for Experiment #2.

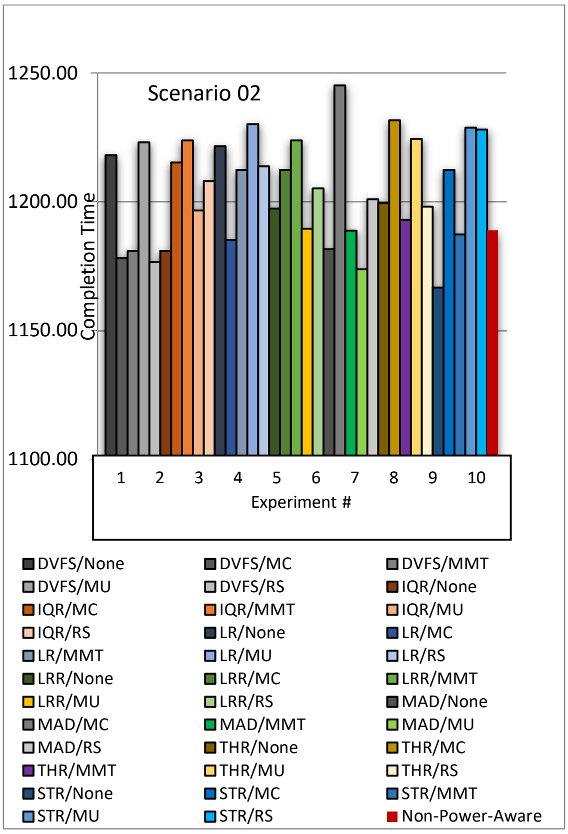

Fig. 6. Scenario 2: Experiment #2 for job completion time.

C. Scenario 3: Increasing the No. of Hosts and Cloudlets

In this scenario, we varied the number of hosts and cloudlets while keeping the number of VMs constant and observed the power consumption and job completion time. The objective of this scenario is to observe the effect of the variation in the number of hosts and cloudlets. The results for the measured power consumption are as follows in Fig. 7. All the DVFS power mechanisms had the least power consumption compared to other policies. Non-power-aware policies have high power consumption concerning all other policies.

Fig. 8 shows the measured results of job completion time. The DVFS/MMT (1172.24 s) had the best execution time. Measured job completion time by all other policies are close to each other, including the non-power-aware policy. The maximum time consumed is IQR/None policy (1250.81 s).

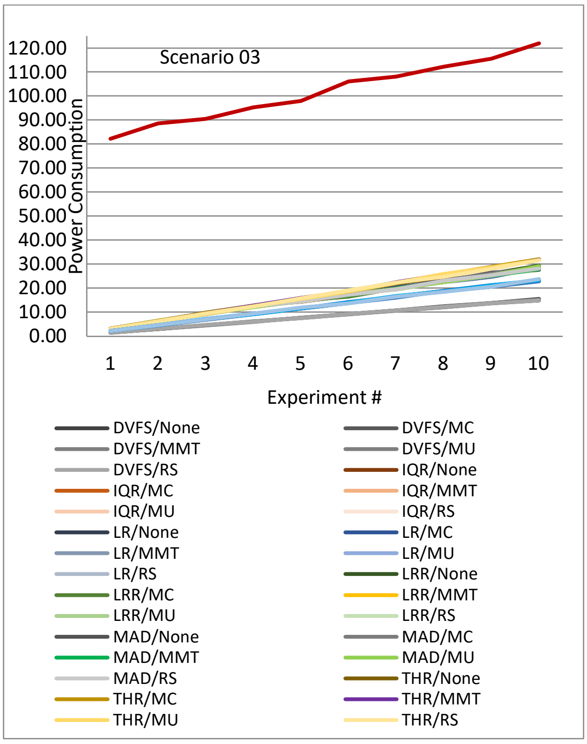

Fig. 7. Scenario 3: Power consumption for Experiment #3.

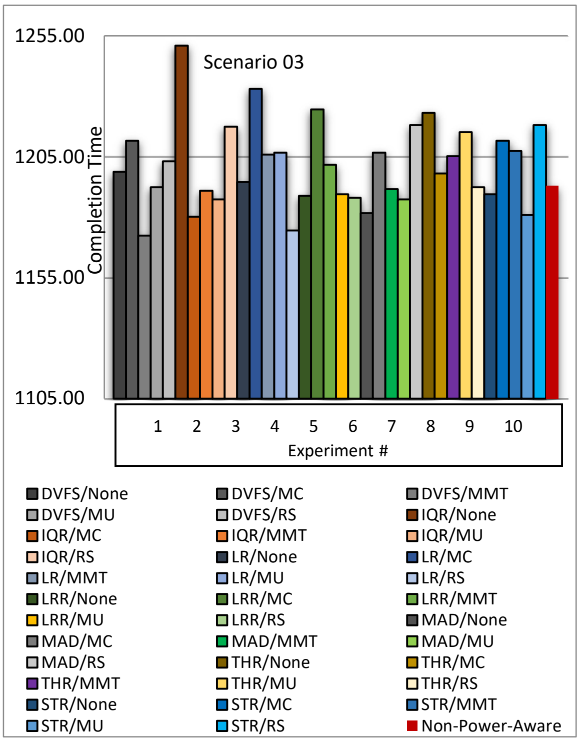

Fig. 8. Scenario 3: Experiment #3 for job completion time.

D. Scenario 4: Increasing the No. of Hosts

In this scenario, we varied the number of hosts while keeping the number of VMs and cloudlets constant and observed the power consumption and job completion time. The objective was to observe the effect of the variation in the number of hosts.

The results for the power consumption in Fig. 9 are as follows. The power consumption remained constant except for the Non-Power-Aware because the new hosts that were added were not utilized. All the DVFS power mechanisms had the least power consumption in the 15.10 to 15.37 W range.

Fig. 10 shows the measured results for job execution time. The STR/None had the best execution time (1175.10 s). Even though the non-power-aware had a competitive job completion time, its power consumption was the worst, as it kept all the servers powered on.

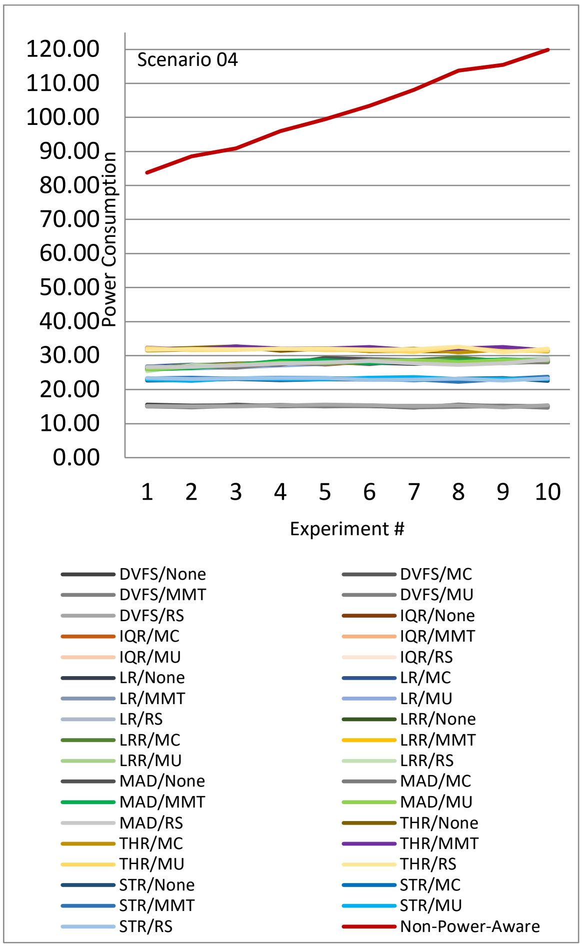

Fig. 9. Scenario 4: Power consumption for Experiment #4.

E. Scenario 5: Increasing the No. of VMs and Cloudlets

In this scenario, we varied the number of VMS and Cloudlets while keeping the number of hosts constant and observed the power consumption and job completion time. The objective of this scenario is to observe the effect of the variation in the number of VMs and Cloudlets.

The results for the power consumption are as follows (Fig. 11). The Power consumption remained constant (83.50–84.53 W) in the None-Power-Aware because the number of hosts remained constant. All the DVFS power mechanisms had the least power consumption concerning all investigated policies.

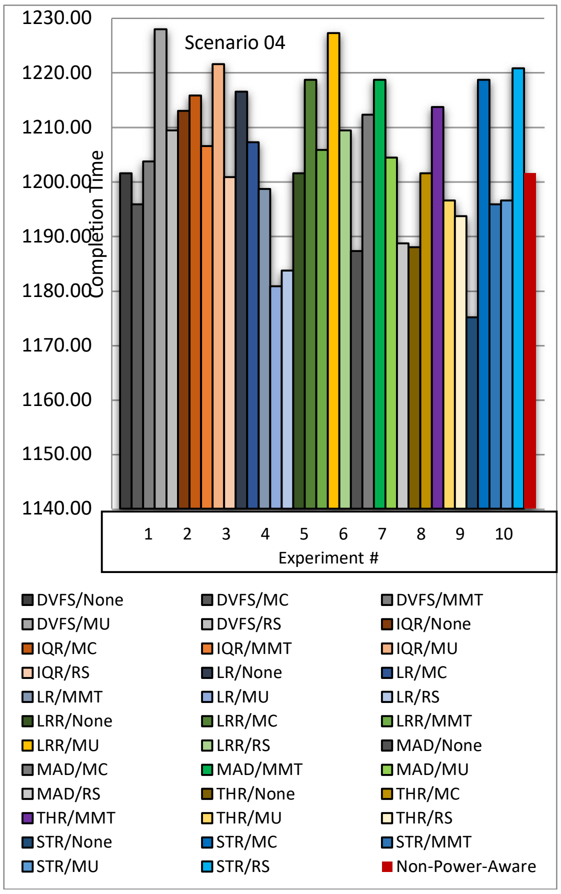

Fig. 10. Scenario 4: Experiment #4 for job completion time.

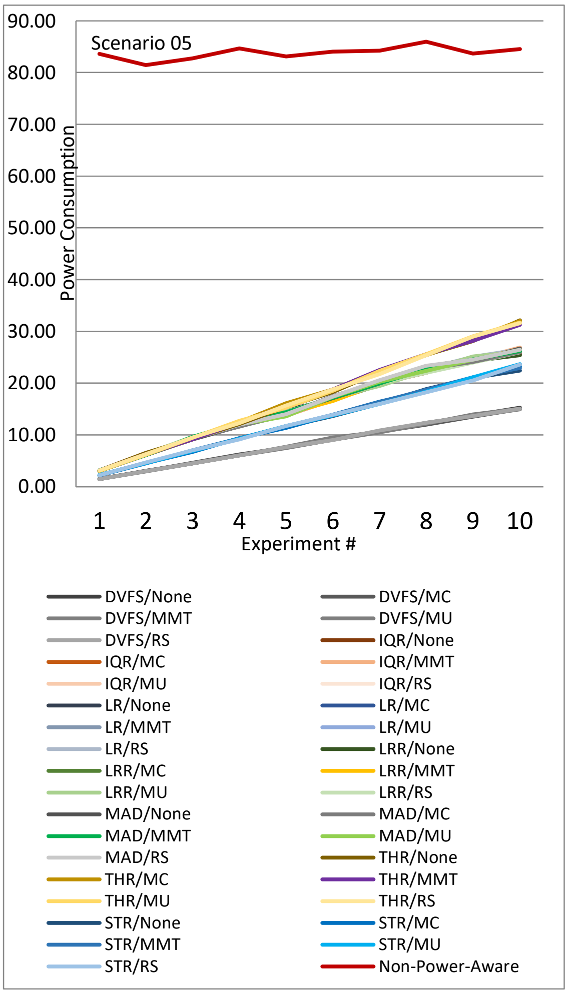

Fig. 11. Scenario 5: Power consumption for Experiment #5.

Fig. 12 shows the measured job completion time under this scenario. The MAD/MC had the best execution time (1166.53 s). The STR/MC takes the maximum job completion time (1229.39 s). The non-power-aware policy takes 1219.39 s, comparable and close to the minimum execution time.

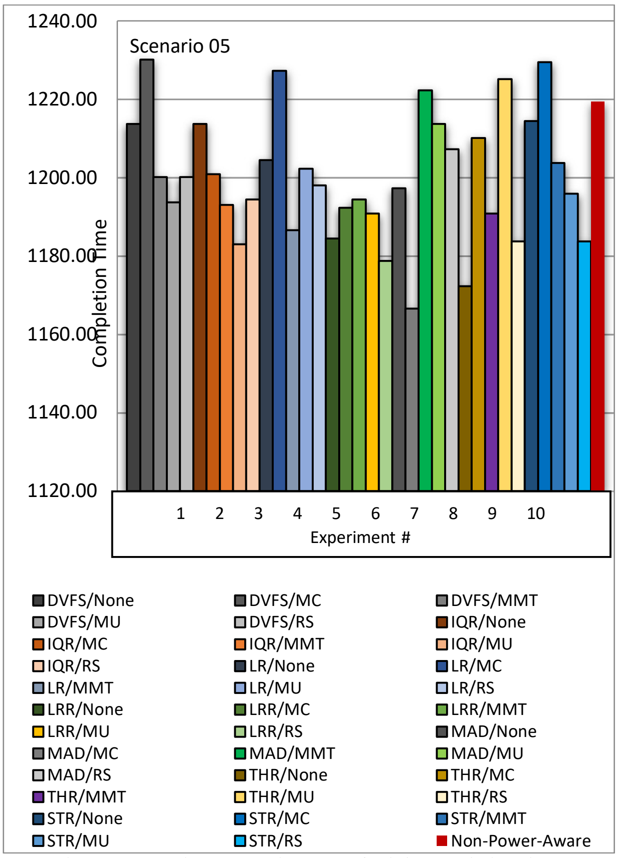

Fig. 12. Scenario 5: Experiment #5 for job completion time.

A. Scenario 6: Increasing the No. of VMs

In this scenario, we varied the number of VMs while keeping the number of hosts and cloudlets constant and observed the power consumption and job completion time. The objective of this scenario is to observe the effect of the variation in the number of VMs.

Fig. 13 shows the measured results for power consumption under this scenario. Power consumption remained constant in the None-Power-Aware because the number of hosts remained constant (82.75–82.94 W). All the DVFS power mechanisms consumed the least power in the 2.23–15.31 W range.

The measured job completion time is presented in Fig. 14. These results show that THR/RS had the best execution time (1166.53 s). The job completion time achieved by STR/MU is maximum (1231.53 s). The non-power-aware policy achieved a job completion time of 1209.39 s, which is very close to the job completion time achieved by THR/RS.

B. Scenario 7: Increasing the No. Cloudlets

In this scenario, we varied the number of cloudlets while keeping the number of hosts and

VMs were constant, and the power consumption and job completion time were observed. The objective of this scenario is to observe the effect of the variation in the number of cloudlets.

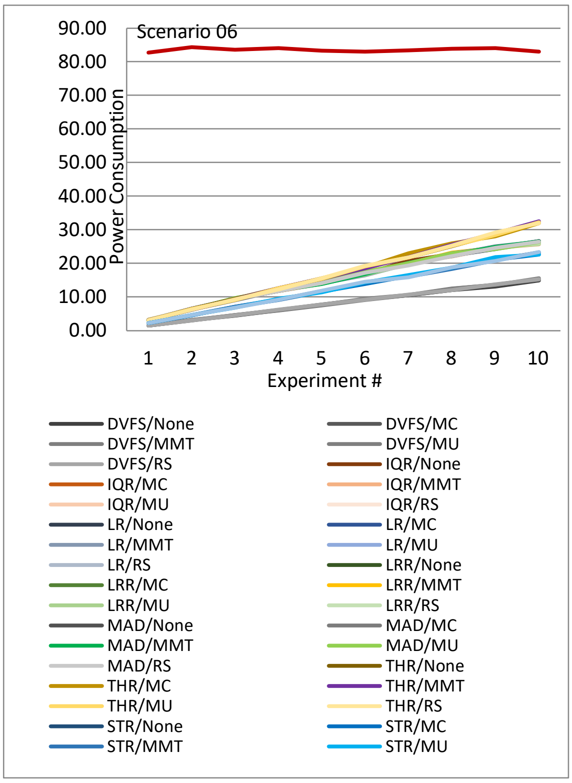

Fig. 13. Scenario 6: Power consumption for Experiment #6.

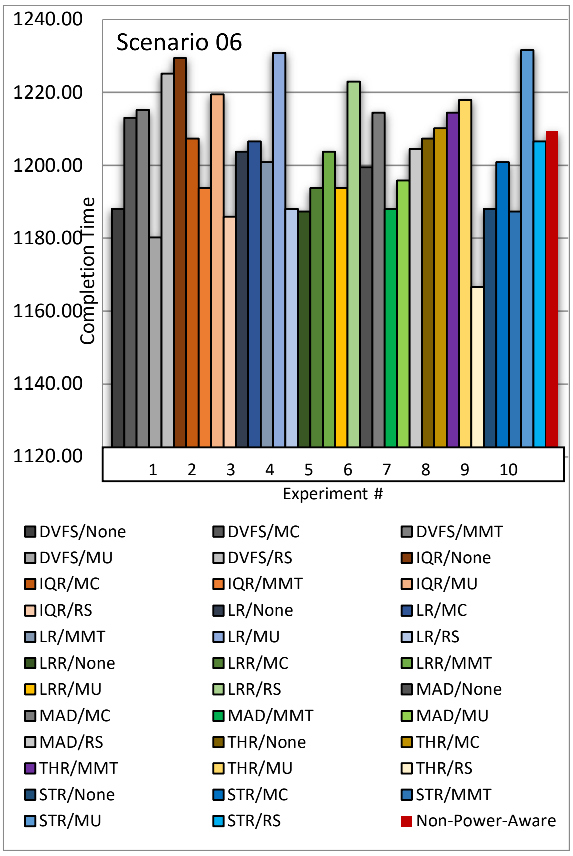

Fig. 14. Scenario 6: Experiment #6 for job completion time.

The results for the power consumption are shown in Fig. 15. The None-Power-Aware's Power consumption remained constant because the number of hosts remained constant (84.13–84.28 W). All the DVFS power mechanisms had the least power consumption. The measured job completion time is listed in Fig. 16. The MAD/MMT had the best execution time.

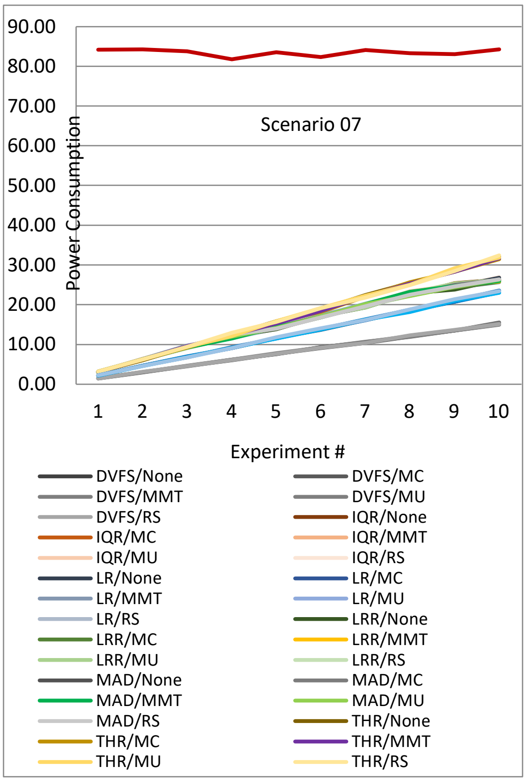

Fig. 15. Scenario 7: Power consumption for Experiment #7.

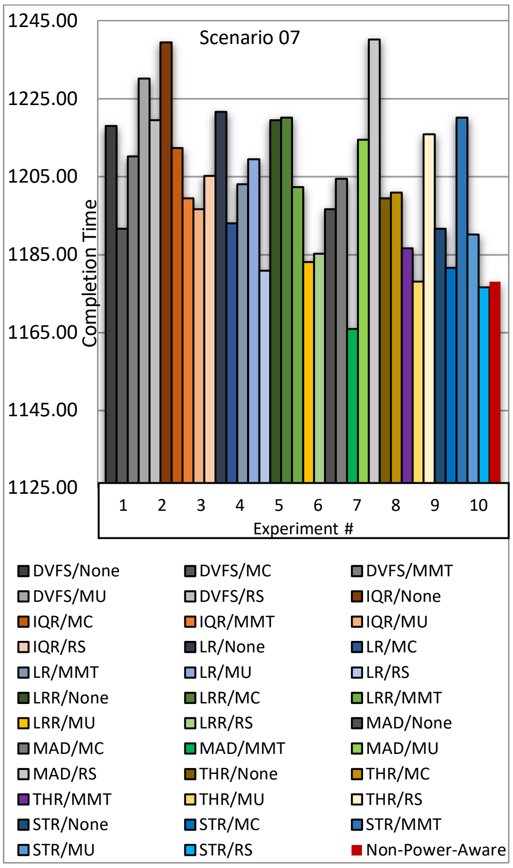

Fig. 16. Scenario 7: Experiment #7 for job completion time.

VI. Overall Discussion and Results Comparison

In this paper, we have identified that specific policies prioritize power consumption rate while others prioritize job completion rate. The DVFS mechanism has proven to be the most effective method for conserving power. However, the DVFS mechanism doesn't always perform the best regarding job completion time. Therefore, considering both power consumption and job completion time, it is essential to consider the priorities of the jobs that need to be executed. This will help the users choose the policy that meets their requirements.

VII. Conclusion and Future Direction

The limitation of this research is that simulation tools conduct the experiments. The research findings will be appreciated if the data and the experiments are in real-time. In addition to power consumption, green computing involves more sensitive issues, such as carbon emission and electronic waste. Potential future research is to consider these parameters of green computing while assessing the computing efficiency of cloud infrastructures.

As we mentioned, Cloud Technology is an unavoidable system in large enterprises. The price for Cloud Technology is the compromise of the values of green computing. Green computing is bundled with a performance-energy trade-off. It is the job completion rate versus the power consumption rate. With the growing use of Cloud Computing, it has become increasingly essential to manage power consumption and execution time for jobs. Based on the analysis of seven scenarios (Figs. 3, 5, 7, 9, 11, 13, and 15), it has been observed that DVFS (Dynamic Voltage and Frequency Scaling) demonstrates the best power consumption compared to the other six methods. Table 6 provides a summary of the measured results. The job completion time measurement in Figs. 4, 6, 8, 10, 12, 14, and 16 indicates that the combination of THR/MU and THR/MMT yields the best job completion time. Furthermore, the performance of the various combinations of power consumption policy and VM allocation policy is prioritized accordingly. Table 6 shows the summary of measured results in scenario 1–7 in term of best in power consumption and best in job completion time.

Table 6. Summary of experimental results from Scenario 1–7

Scenario Number	Best in Power Consumption	Best in Job Completion Time
1	DVFS	THE/MU, THR/MMT
2	DVFS	STR/None
3	DVFS	DVFS/MMT
4	DVFS	STR/None
5	DVFS	MAD/MC
6	DVFS	THR/RS
7	DVFS	MAD/MMT

The measured results in this investigation were obtained using the CloudSim simulator; however, the results need to be validated in a real cloud computing environment. Another area that could be explored in the future is measuring cost and benefit analyses from cloud users, and cloud managers' perspective is to refer to [16] and [17].

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

This research work is an outcome of MS-Project which is supervised by Dr. Kalim Qureshi. The implementation and measurement work is done by Mr. Abdulatif Albusairi. The data analysis work, formatting and English correction is done jointly by all authors. All authors had approved the final version.

References

[1] A. A. Khan, M. Zakarya, I. U. Rahman, R. Khan, and R. Buyya, "An energy and performance efficient resource orchestrator for hybrid heterogeneous cloud computing environments," *Journal of Network and Computer Applications*, vol. 173, pp. 1–27, 2021.

[2] N. McDonnell, E. Howley, and J. Duggan, "Dynamic virtual machine consolidation using a multi-agent system to optimize energy efficiency in cloud computing," *Future Gener Comput. Syst*., vol. 108, pp. 288–301, 2020.

[3] N. K. Sharma, S. Bojjagani, Y. C. A. P. Reddy, M. Vivekanandan, J. Srinivasan, and A. K. Maurya, "A novel energy efficient multi-dimensional virtual machines allocation and migration at the cloud data center," *IEEE Access*, pp. 107480–107495, 2023.

[4] R. Sharma, S. Nikam, and A. Manjrekar, "Green cloud computing: Challenges and solutions," *European Chemical Bulletin*, vol. 12, no. 5, pp. 6828–6831, 2023. doi: 10.48047/ecb/2023.12.si5a.0616

[5] S. Bhattacherjee, R. Das, S. Khatua, and S. Roy, "Energy-efficient migration techniques for cloud environment: A step toward green computing," *The Journal of Supercomputing*, vol. 76, pp. 5192–5220, 2020. https://doi.org/10.1007/s11227-019-02801-0

[6] J. Park, K. Han, and B. Lee, "Green cloud? An empirical analysis of cloud computing and energy efficiency," *Management Science*, vol. 69, no. 3, 2022. https://doi.org/10.1287/mnsc.2022.4442

[7] S. Farzai, M. H. Shirvani, and M. Rabbani, "Multi-objective communication-aware optimization for virtual machine placement in cloud datacenters", *Sustainable Computing: Informatics and Systems*, vol. 28, 100374, 2020.

[8] R. Boutaba, Q. Zhang, and M. F. Zhani, "Virtual machine migration in cloud computing environments: Benefits, challenges, and approaches," in *Advances in Systems Analysis, Software Engineering, and High-Performance Computing (ASAEHPC) Book Series*, 2013.

[9] X. Liu, J. Wu, G. Sha, and S. Liu, "Virtual machine consolidation with minimization of migration thrashing for cloud data centers", *Mathematical Problems in Engineering*, pp. 11–50, 2022.

[10] R. Chen, B. Liu, W. Lin, J. Lin, H. W. Cheng, and K. Li, "Power and thermal-aware virtual machine scheduling optimization in cloud data centre," *Future Generation Computer Systems*, vol. 145, pp. 578–589, 2023.

[11] U. Arshad, M. Aleem, G. Srivastava, and J. C.-W. Lin, "Utilizing power consumption and SLA violations using dynamic VM consolidation in cloud data centers," *Renew Sustain. Energy Rev*., vol. 167, 112782, pp. 1–14, 2022.

[12] J. Wang, H. Gu, J. Yu, Y. Song, X. He, and Y. Song, "Research on virtual machine consolidation strategy based on combined prediction and energy-aware in cloud computing platform," *Journal of Cloud Computing*, vol. 50, pp. 1–18, 2022.

[13] Y. Saadi and S. El-Kafhali, "Energy-efficient strategy for virtual machine consolidation in cloud environment," *Soft Comput*., vol. 24, no. 19, pp. 14845–14859, 2020.

[14] S. Mangalampalli, G. R. Karri, and K. V. Rajkumar, "EVMPCSA: Efficient VM packing mechanism in cloud computing using chaotic social spider algorithm," *Procedia Comput. Sci*., vol. 218, pp. 554–562, 2023.

[15] M. Rezakhani, N. Sarrafzadeh-Ghadimi, R. Entezari-Maleki, L. Sousa, and A. Movaghar, "An energy-aware QoS-based dynamic virtual machine consolidation approach based on RL and ANN," *Cluster Comput*., vol. 27, no. 1, pp. 827–843, 2024.

[16] M. Sahinoglu, S. Morton, C. Vadla, and K. Medesani, *CLOUD Risk Assessment & Management Survey Tool for Users and Providers*, Dusseldorf, Germany: LAMBERT Academic Publishing, 2017.

[17] M. Sahinoglu, *Cyber-Risk Informatics: Engineering Evaluation with Data Science*, Hershey PA: John Wiley & Sons, 2016, pp. 383–408.